COMPUTER GRAPHICS IN COMPUTER SCIENCE EDUCATION

Jezek Bruno, Vanek Jan,
University of Defence, Hradec Kralove, Czech Republic
E-mail: jezek@pmfhk.cz, vanek@pmfhk.cz

Slaby Antonin
University of Hradec Kralove, Hradec Kralove, Czech Republic
E-mail: antonin.slaby@uhk.cz

Abstract

For the past several years computer graphics subjects have played an important role in information and computer technology studies. The inclusion of computer graphics subjects into university curricula is motivated not only by modern technologies and methods being used in this branch, but above all by the training of programming techniques and math application these subjects provide. At the Faculty of Informatics and Management of the University of Hradec Kralove, subjects Computer Graphics I, II and III (CG) have been professed for several years within the curricula of the Applied Informatics and the Information Management study programs. Besides visualisation and graphical data processing, these subjects are focused on practicing and widening the knowledge and practical skills of students, future IT specialists. This paper introduces goals, structure and educational techniques in the CG subjects, which expand informatics subjects such as programming and algorithm development. Educational methods that involve a term project, a special framework and an e-learning support are also described. Finally several specific examples of math application used during problem solving are presented.

Key words: *computer graphics, programming education, algorithm and math application*.

Introduction

For the past several years modern computer technologies have become an integral part of our life and this has imposed greater demands on students' knowledge in the field of information and communication technology. New subjects improving software and hardware usage and design skills are added to university curricula. Besides established subjects such as algorithm development, computer programming, object technologies, information systems and others, a subject, which deals with computer graphics, appears in a number of study programs. The inclusion of computer graphics subjects into university curricula is motivated not only by modern technologies and methods being used in this branch, but above all by the training of programming techniques and math application these subjects provide.

At the Faculty of Informatics and Management of the University of Hradec Králové (FIM UHK), subjects Computer Graphics I, II and III (CG) have been professed for several years within the curricula of the Applied Informatics and the Information Management study programs (Slaby et al, 2005). Besides visualisation and graphical data processing, these subjects are focused on practicing and widening the knowledge and practical skills of students, future IT specialists.

Goals

In spite of the fact that computer graphics is currently a modern and attractive branch the percentage of students, who will be dealing with it within their professional practice, is relatively small. Only few students will find jobs as developers of CAD or DTP graphical systems or will create visualization cores of game engines for example. Most of the graduates will be engaged in standard tasks that require higher levels of programming skills and knowledge, but will not solve special problems of computer graphics.

Then, why are computer graphics subjects compulsory? The problems solved in the computer graphics enable us to practise and deepen the knowledge and skills that students obtained in the area of programming (Kozel, 2009) and algorithm development (Milkova et al, 2007) and simultaneously teach students to apply mathematic tools gained in theoretical subjects. The solved tasks utilize mathematic knowledge and force students to suggest and correctly implement a software solution. Created programs have mainly graphical output which gives both the students and their lecturers immediate answers to questions such as whether the algorithm is correctly implemented or in which part of the algorithm and its implementation a mistake occurred. An additional advantage lies usually in an attractive formulation of the projects to be solved by the students and very impressive results, which students obtain by dealing with them. All of this attracts the students and stimulates them to further work.

Organisation of the subject

The education in computer graphics at the Faculty of Informatics and Management is divided into three subjects of Bachelor's and Master's degree programmes. The Computer Graphics I and the Computer Graphics II are taught in the second year of Bachelor's studies. The Computer Graphics III is lectured in the first year of the follow-up Master's degree programme. All subjects are compulsory for students of Applied Informatics and optional for students of other branches.

Computer Graphics I

The subject Computer Graphics I is mainly focused on basic visualization algorithms, both planar and spatial. The lectured procedures and algorithms are thoroughly practiced on a series of simple sequential problems having a scope of one or two seminars each. Students are not restricted to the use of a specific programming language and development tool. They usually choose Java and the Eclipse environment (Eclipse, 2009), some use C++ as their programming language of choice and develop in Microsoft Visual Studio. Within the scope of the seminars of the subject Computer Graphics I no special graphical libraries are used and a function to render a single simplest graphical element (pixel) is virtually the only graphical function available. Further more complicated functions for plotting of segments, polygons or for object transformations are implemented by the students themselves. Later during the term students get a source code of vector and matrix maths libraries, which are used for planar and spatial transformations, projections and a 3D scene viewer set up.

Computer Graphics II

In the second term (in the scope of the subject Computer Graphics II) the theoretical knowledge is extended to the area of advanced visualisation algorithms, commonly used graphical models, graphical formats, computer-aided recognition and computer vision. In this subject the students are shown a huge range of applications that use computer processing and transformation of graphical information. The students themselves then decide the area in which they will be involved in their main term project. Besides the main task, the students work on two minor tasks during the seminars. The first minor task is oriented towards implementation of one of visibility solving algorithm used for

3D scene visualization. Students have to use their knowledge from the previous term and implement several partial algorithms solving the requested task. In this way the students further improve their knowledge and skills of algorithm implementation, data structure usage, object oriented programming and other abilities. They gain the capability to manage and debug a large scale software project, detect programme code errors and correct them in case of need, and learn to trace the program and the value of variables.

The second minor task is focused on familiarisation with a professional graphical library, for example OpenGL. Students get acquainted with functions that implement already known graphical algorithms and data structures. They are given some samples of the usage of these functions, based on which they solve simple visualisation problems and investigate the consequences of setting various parameters of the functions.

Computer Graphics III

The youngest of computer graphics subjects at FIM is Computer Graphics III lectured during winter term of the first year of the follow-up Master‘s degree programme. In the follow-up programmes the students already have a good basic knowledge in the area and, additionally, they have successfully defended their bachelor's theses. Therefore, the subject is focused on some advanced algorithms and new software and hardware architectures used in modern graphics cards. This technology uses special programmable processor units called shaders placed in certain stages of visualization pipeline. Students get familiar with functions of vertex, pixel and geometry shaders, with their programming capabilities and with algorithms used in these technologies. Real hardware and a software simulation framework are employed for implementation of more complicated visualization techniques such as parallax mapping.

Bachelor's and Mater's theses

Some students get so involved in the problems of computer graphics that they want to solve substantially more complicated and complex tasks then those that are offered to them in the list of the main term project topics. Sometimes, such students extend the tasks and continue solving them within their Bachelor's and Master's theses. They form a group in which they, together with their lecturers, discuss the problems they are dealing with and familiarise others with new findings by giving presentations at the regular meetings of the group. This computer graphics group is supported by the Faculty of Informatics and Management, which provides hardware and software equipment and financially contributes to students' participations at scientific conferences.

Educational methods

As can be seen above, the computer graphics subjects are focused on practising of programming skills and deepening abilities to apply theoretical knowledge obtained in other areas. The main educational methods, which help achieve this goal, are mentioned in the following paragraphs.

Main term project and scientific paper

The students use all the acquired knowledge to solve the main term project, which they work on during the second half of the term. The expected output of the main term project is composed of three parts: working software application, presentation and scientific paper describing solved problem. Students have to deliver all of them. The software includes a functional program (developed by students in their chosen programming language) and its source code with appropriate comments. The presentation is given during the last seminars of the term and is followed by a discussion. The students are required to create the project documentation in the form of a scientific paper with all the proper parts – abstract, introduction, methods, implementation, results and conclusion. The topics of term projects for students to choose from were selected in order to force students to study the problem area more deeply then it is dealt with in lectures. They learn to work with generally available internet sources and on-line digital libraries, such as IEEE and ACM, which are accessible at the FIM. The paper is an important component of the main project because it trains students in methods of scientific writing, which they would later take advantage of in their Bachelors' and Masters' theses. Students are provided with a template of the sample paper to assist them in their own work. Students are not expected to write a real scientific paper, but they should play to be scientists and try to formulate studied ideas.

The main term project means a climax of the students endeavour and for some of them it is their last work in the area of computer graphics. The students try to design and implement a large scale project concentrated on a specific part of the lectured area of computer graphics. Students can choose from many problems which include serious practical applications as well as impressive trifles (Figure 1). Some of them are for example 3D maze, oscillating spiral spring, rippling water surface, 3D view of the solar system, recognition of simple objects (dice) and many others. In the beginning some students are almost terrified by the complexity of the tasks but having analysed the problem and recognised that they had possessed all the necessary information they usually succeed in solving the chosen problem.

Framework

No special software environments are prepared in seminars of Computer Graphics I and II subjects. Students only have the already mentioned software packages with classes of vector and matrix mathematic operations and some special classes for cubics and quaternions. These libraries help students to implement mathematic operations without time-consuming programming and debugging. In the subject Computer graphics III it is different. Students use a framework for simulation of programmable visualization pipeline based on shader architecture as it is implemented in modern graphics cards. The framework was first introduced at the time when the shader architecture was being lectured, but classrooms lacked computers with appropriate hardware. However, it turned out, that the advantages of the framework go beyond that and that students understand a structure, functionality and data flows of the whole shader architecture. The software solution implemented in Java programming language is of course slower, but after familiarizing with principles students can code into shader programmes and launch on appropriate hardware.

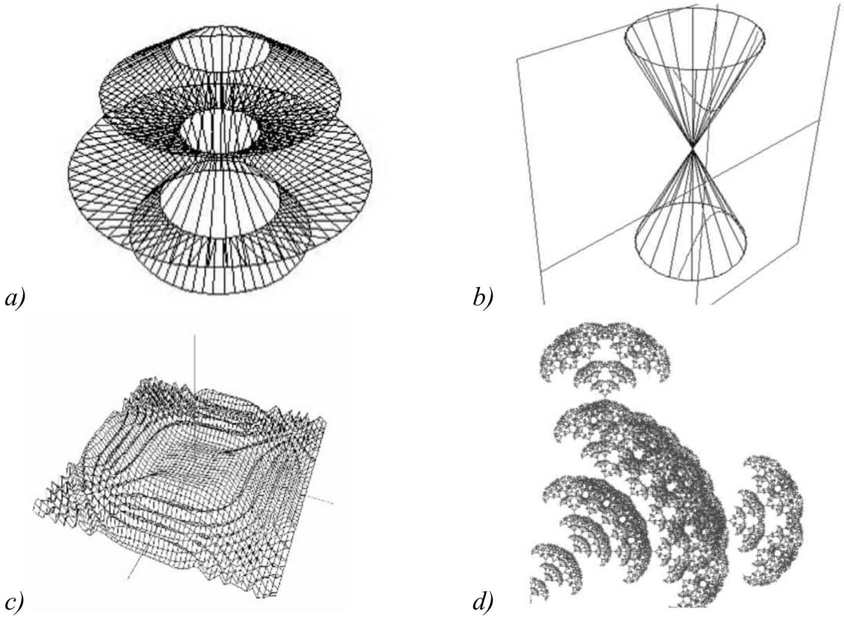

Figure 1. Samples of graphical results of some of CGI term projects: a) rotation pattern, b) conic section, c) function visualization, d) L-systems.

GLSL

The graphics library OpenGL has been lectured and trained for many years especially within the scope of the subject Computer graphics II. With the ongoing shader technology development the lectures were expanded with shader programming language. In regard of students' previous experience with OpenGL the Graphics library shading language (GLSL) has been chosen. A freeware program ShaderDesigner (Shader Designer, 2009) is used for presentation of methods of visualization and illumination of a 3D scene. This software enables user-friendly programming of both vertex and fragment shader programs and setting up a graphics scene. We use ShaderDesigner to show lectured algorithms and methods in real-time, because it allows us to immediately modify parameters or implemented procedures. Plenty of algorithms can be debugged before they become part of final application.

E-learning

Even though this paper is not concerned with using e-learning methods, it is necessary to shortly mention this education support too. FIM UHK has a long tradition of using e-learning in education. The WebCT learning management system (WebCT, 2009) has been successfully employed for several years. Not only does WebCT provided study materials, but it also enables students and teachers to communicate and it manages students' tasks and assessments. The built-in calendar clearly defines deadlines and spares teachers pointless discussions about students' unverifiable claims. It is also quite easy for teachers to edit educational contents, create tests and manage individual study groups. Students usually appreciate services of the WebCT system, because they can access it from anywhere they are.

Cases

The following specific cases present several examples of tasks solved in computer graphics subjects. They also show usage of students' knowledge obtained in other studied subjects and deepening of programming skills.

Implementation and debugging a visibility algorithm

Students have got knowledge of algorithm development from previous informatics subjects. In these subjects there is usually not enough time to finalize a large project due to the amount of subject matter. Solved projects in computer graphics subjects teach students not only to design and implement software solution but also detect code errors and correct them. Besides standard software tools such as debugger, the graphical output of an unfinished project can help localize bugs in program code. This can be demonstrated on the depth buffer visibility algorithm. This algorithm is typically implemented in hardware of modern graphical cards. It could seem that a software implementation gives students nothing new. Both student and teachers cab easily check the correct implementation of transformation application, rasterization, edge rendering and surface illumination calculation. If a student applies transformation (of an object, of a viewer's position or of a scene projection) in an incorrect order, faults in a final image of wireframe model are seen directly, for example in erroneous perspective, rotation or composition of transformation. The rasterization of model faces is also easily checked, because empty or partially filled triangles can be detected immediately. Wrong implementation of the visibility algorithm is apparent on edges of triangles, when parts of back-faces are visible (Figure 2).

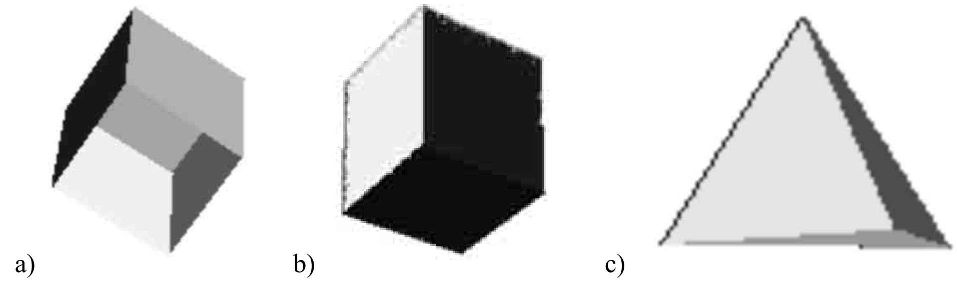

Figure 2. **Samples of the most often students' implementation mistakes a) wrong z-buffer condition, b) incorrect rasterization, c) rendering of back-face's edges.**

Math application in computer graphics

Analysis, linear algebra and other math fields are often used in problem solution and algorithm implementation. Students have the possibility to refresh or deepen their knowledge gained during their university and sometimes even their high school studies. A big advantage of computer graphics is that students can realize practical meanings of math theories and they learn to use it. Typical problems of using math are described in the following text.

Of course all students know terms such as vector and matrix. Students have some notion of geometry theory based on vectors and matrix operations from previous studies. Most of the students are able to calculate dot or cross product of planar or spatial vectors. They are also able to multiply matrixes. But usually they don't know the meaning and consequences of these operations. They can realize the meaning of dot product in the calculation of an angle between vectors or the meaning of cross product in determination of normal vector or area of a triangle. During explanation of matrix usage for transformation of point coordinates the fact of invalidity of the commutative law of multiplication is emphasised. The possibility of using column and row vectors to define homogenous coordinates and correct order of multiplication is also mentioned (Figure 3).

$$\begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix} = \begin{pmatrix} m_{11} & m_{21} & m_{31} & m_{41} \\ m_{12} & m_{22} & m_{32} & m_{42} \\ m_{13} & m_{23} & m_{33} & m_{43} \\ m_{14} & m_{24} & m_{34} & m_{44} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \qquad (x', y', z', w') = (x, y, z, w) \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{pmatrix}$$

$$\mathbf{b} = \mathbf{aT} \qquad \mathbf{b}^{\mathrm{T}} = \mathbf{T}^{\mathrm{T}}\mathbf{a}^{\mathrm{T}}$$

$$\mathbf{b} = \mathbf{aT}_1\mathbf{T}_2 \qquad \mathbf{b}^{\mathrm{T}} = \mathbf{T}_2{}^{\mathrm{T}}\mathbf{T}_1{}^{\mathrm{T}}\mathbf{a}^{\mathrm{T}}$$

Figure 3. **Two ways of vector and matrix multiplication while 3D points are transformed (a, b – column vectors defining 3D point in homogenous coordinates, T – transform matrix 4x4)**

The wide area of math analysis is exercised in tasks of modelling and rendering of curves and surfaces. Students have to know how to define function in parametric or explicit form and also how to interpolate values between computed points. In case of triangulated surfaces students must calculate normal vectors to define a colour or illumination by some shading model (Figure 4).

$$\vec{\mathbf{n}} = \vec{\mathbf{u}} \times \vec{\mathbf{v}}$$

$$\vec{\mathbf{u}} = \left(\frac{\partial \mathrm{f_x}(u,v)}{\partial u}, \frac{\partial \mathrm{f_y}(u,v)}{\partial u}, \frac{\partial \mathrm{f_z}(u,v)}{\partial u} \right)$$

$$\vec{\mathbf{v}} = \left(\frac{\partial \mathrm{f_x}(u,v)}{\partial v}, \frac{\partial \mathrm{f_y}(u,v)}{\partial v}, \frac{\partial \mathrm{f_z}(u,v)}{\partial v} \right)$$

Figure 4. Normal vector determination by cross product of tangent vectors calculated by partial derivation of parametric function.

There are plenty of other fields of math used in computer graphics. Let us name those that are taught in our subjects. Surface intersection determination is used in collision detection and ray tracing. Quaternions are used as an alternative to matrices for rotation definition and application. Convolution and mathematic morphology operations are used in image processing. Probability and theory of information are used in image format compression. Integration of values along ray is used in volume data visualisation.

Results

We have applied the presented practices and methods in computer graphics subjects for more then ten years. We also tried to obtain a feedback and for that students have the possibility to express their feelings in the text part of the main term project. Generally students rate the computer graphics subject as one of the most demanding ones. On the other hand, they value the possibility to increase their ability to finalize software project and broaden their experience with programming. The following list summarizes skills that students develop at various stages of problem solving process.

- stage 1: preparation for the solution
  - analysis of the problem
  - application of theoretical knowledge
  - work with software help and internet resources
  - searching for relevant papers in online journal databases
  - ability to understand and use the text written in a foreign language
- stage 2: implementation of the solution
  - algorithm development
  - design of data structures
  - programming and documentation of the program
  - debugging, profiling, correcting errors
  - improvement of the designed approach and widening and generalisation of the problem
- stage 3: presentation, discussion, defence
  - formulation of the text of the documentation
  - making graphical presentation to introduce a problem and a solution in limited time
  - presentation of the problem and the students' own method of solution
  - conclude own solution and note its advantages and disadvantages

We observe a continual increase of students' ability to present their own solutions in both forms – a short talk and a scientific paper. Students use this acquired experience also in their Bachelor's and Master's theses. Every year students' diploma theses are rated in the faculty competition and those with computer graphics topics are usually among winners.

Conclusion

Our approaches and methods of computer graphics education turned out to be efficient. By solving various tasks in the computer graphics subjects, all the students improved their programming and algorithm development skills and overall information technology knowledge. We have also managed to increase the students' interest in programming and computer graphics. The enthusiasm aroused in some students led to the creation of very successful theses. We mention below the feedback obtained from text parts of several of the main term projects.

"… The subject Computer Graphics I is one of the most demanding but also among my most favourite subjects. It helped me to understand 3D applications, principles of line, polygon, 2d and 3D objects drawing. Besides that, I learned programming more than in all the other programming subjects together. I repeated math that I missed in this term. Thank you …"

"… Finally, I would like to remark that, in spite of the fact that I had not been extremely interested in computer graphics, in the end I liked both the subject and the tasks. I also gradually started to prefer computer graphics to other more complicated and urgent subjects. Every result I obtained brought a great pleasure to me. Viewing 3D scene is more exciting for me than a grey window of database application …"

"…I rate the subject very positively and I think that it is useful for students. They learn the right programming habits and basic math principles of 2D and 3D computer graphics. Paper writing is also a good training for theses. I was satisfied with seminars and I appreciated solving of practical topics…"

These and similar remarks are rather common and typical. At the beginning students show some hesitation towards computer graphics but gradually it becomes a favourite subject for them. They find the solution of graphical problems to be a meaningful task producing impressive results, suitable for presentation. During the solution of the problems students often do not realise the fact that they work on a task aimed only at extending and deepening their abilities in applying information technology.

In the future we want to continue in the approach established in computer graphics subjects and to expand and enrich lectured topics. We further improve the study materials and demonstration samples. The ever increasing complexity of computer graphics algorithms gives these subjects a good potential to improve students' skills in the future as well.

Acknowledgements

The paper was prepared with the support of grant FVZ UO MO 9079301011033.

References

Eclipse. (2009). Retrieved February 1, 2009, from http://www.eclipse.org/

Kozel, T. (2009). Programmimg I, II, III. Retrieved February 1, 2009, from http://iris.uhk.cz/kozel/

Milková, E. Kozel, T. (2007). Od algoritmizace k programování. *Informatika XX/2007* Brno: Konvoj, 2007, 1-6.

Shader Designer. (2009). Retrieved February 1, 2009, from http://www.typhoonlabs.com/

Slabý A., Ježek B., Vaněk J. (2005). Using computer graphics to improve programming skills. *International*

Journal of Continuing Engineering Education and Life-Long Learning, Vol. 15, No. 3/4, 185-193.

WebCT. (2009). Retrieved February 1, 2009, from http://www.blackboard.com/

Adviced by Eva Milkova,
University of Hradec Kralove, Hradec Kralove, Czech Republic

Jezek Bruno PhD., Lecturer at Faculty of Military Health Sciences, University of Defence, Trebesska 1575, Hradec Kralove, Czech Republic
E-mail: jezek@pmfhk.cz
Website: http://www.pmfhk.cz

Vanek Jan Lecturer at Faculty of Military Health Sciences, University of Defence, Trebesska 1575, Hradec Kralove, Czech Republic.
E-mail: vanek@pmfhk.cz
Website: http://www.pmfhk.cz

Slaby Antonin Professor, Faculty of Informatics and Management, University of Hradec Kralove, Rokitanskeho 62, 500 03 Hradec Kralove, Czech Republic.
E-mail: antonin.slaby@uhk.cz
Website: http://www.uhk.cz/eng/